Kernel Regression with Order Preferences

Xiaojin Zhu Andrew B. Goldberg

Department of Computer Sciences
University of Wisconsin, Madison, USA

AAAI 2007

How much is your house worth?

Regression problem.

- labeled training data $(x_1, y_1), \ldots, (x_l, y_l)$
- $y \in \mathbb{R}$: price
- x: features

How much is your house worth?

Regression problem.

- labeled training data $(x_1, y_1), \ldots, (x_l, y_l)$
- $y \in \mathbb{R}$: price
- x: features
  - location

How much is your house worth?

Regression problem.

- labeled training data $(x_1, y_1), \ldots, (x_l, y_l)$
- $y \in \mathbb{R}$: price
- x: features
  - location
  - location

How much is your house worth?

Regression problem.

- labeled training data $(x_1, y_1), \ldots, (x_l, y_l)$
- $y \in \mathbb{R}$: price
- x: features
  - location
  - location
  - location

How much is your house worth?

Regression problem.

- labeled training data $(x_1, y_1), \ldots, (x_l, y_l)$
- $y \in \mathbb{R}$: price
- x: features
  - location
  - number of bedrooms
  - age
  - median income
  - ...
- learn $f : X \mapsto \mathbb{R}$

Knowledge from real estate experts

"Within some distance, other factors being roughly equal, the value is largely determined by the number of bedrooms."

Order preferences

One way to express the knowledge is to use order preferences on unlabeled data $x_{l+1}, x_{l+2}, \ldots$

For some x_i, x_j, we may not know $f(x_i), f(x_j)$.
But we prefer $f(x_i) \geq f(x_j)$.

Order preferences

One way to express the knowledge is to use order preferences on unlabeled data $x_{l+1}, x_{l+2}, \ldots$

For some x_i, x_j, we may not know $f(x_i), f(x_j)$.
But we prefer $f(x_i) \geq f(x_j)$.

Definition

An order preference is a tuple (i, j, d, w), so we prefer $f(x_i) - f(x_j) \geq d$ with confidence w.

Order preferences can encode various information

order $f(x_i) - f(x_j) \geq d$ (i, j, d, w)

Order preferences can encode various information

order	$f(x_i) - f(x_j) \geq d$	(i, j, d, w)
equal	$f(x_i) = f(x_j)$	$(i, j, 0, w), (j, i, 0, w)$

Order preferences can encode various information

order	$f(x_i) - f(x_j) \geq d$	(i, j, d, w)
equal	$f(x_i) = f(x_j)$	$(i, j, 0, w), (j, i, 0, w)$
close	$\lvert f(x_i) - f(x_j) \rvert \leq \epsilon$	$(i, j, -\epsilon, w), (j, i, -\epsilon, w)$

Order preferences can encode various information

order	$f(x_i) - f(x_j) \geq d$	(i, j, d, w)
equal	$f(x_i) = f(x_j)$	$(i, j, 0, w), (j, i, 0, w)$
close	$\|f(x_i) - f(x_j)\| \leq \epsilon$	$(i, j, -\epsilon, w), (j, i, -\epsilon, w)$
interval	$a \leq f(x_i) - f(x_j) \leq b$	$(i, j, a, w), (j, i, -b, w)$

Order preferences can encode various information

order	$f(x_i) - f(x_j) \geq d$	(i, j, d, w)
equal	$f(x_i) = f(x_j)$	$(i, j, 0, w), (j, i, 0, w)$
close	$\lvert f(x_i) - f(x_j)\rvert \leq \epsilon$	$(i, j, -\epsilon, w), (j, i, -\epsilon, w)$
interval	$a \leq f(x_i) - f(x_j) \leq b$	$(i, j, a, w), (j, i, -b, w)$
unary	$f(x_i) \geq g(x_i)$	special case, $g()$ given

Regression with order preferences

- Given:
  - labeled data $(x_1, y_1), \ldots, (x_l, y_l)$
  - unlabeled data $x_{l+1}, \ldots, x_{l+2p}$
  - order preferences $(i_1, j_1, d_1, w_1), \ldots, (i_p, j_p, d_p, w_p)$
- learn $f : X \mapsto \mathbb{R}$

Standard kernel regression

$$\min_{\mathbf{f}\in\mathcal{H}} \sum_{i=1}^{l} c(x_i, y_i, f(x_i)) + \lambda\Omega(\|\mathbf{f}\|_{\mathcal{H}})$$

- $c()$ loss function
- λ regularization weight
- $\Omega()$ monotonic increasing function

Kernel regression with order preferences

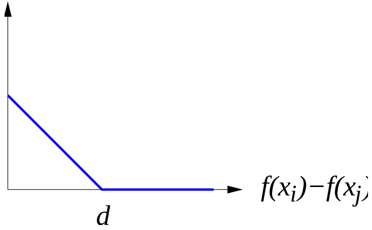
$$\min_{\mathbf{f}\in\mathcal{H}} \sum_{i=1}^{l} c(x_i, y_i, f(x_i)) + \lambda_1 \Omega(\|\mathbf{f}\|_{\mathcal{H}}) + \lambda_2 r(x, f)$$

- $c()$ loss function
- λ_1, λ_2 regularization weight
- $\Omega()$ monotonic increasing function
- $r(x, f)$ order preference regularization

Order preferences as regularization

$$(i, j, d, w) : f(x_i) - f(x_j) \geq d$$

$$w \max(d - (f(x_i) - f(x_j)), 0)$$



$$r(x, f) = \sum_{q=1}^{p} w_q \max(d_q - (f(x_{iq}) - f(x_{jq})), 0)$$

Representer Theorem

The minimizer $\mathbf{f}$ of

$$\min_{\mathbf{f}\in\mathcal{H}} \sum_{i=1}^{l} c(x_i, y_i, f(x_i)) + \lambda_1 \Omega(\|\mathbf{f}\|_{\mathcal{H}}) + \lambda_2 r(x, f)$$

admits the form

$$f(x) = \sum_{i=1}^{l+2p} \alpha_i K(x_i, x)$$

We optimize α.

Design choices

$$\min_{\mathbf{f}\in\mathcal{H}} \sum_{i=1}^{l} c(x_i, y_i, f(x_i)) + \lambda_1 \Omega(\|\mathbf{f}\|_{\mathcal{H}}) + \lambda_2 r(x, f)$$

- Loss $c(x, y, f(x)) \equiv |y - f(x)|$

Design choices

$$\min_{\mathbf{f}\in\mathcal{H}} \sum_{i=1}^{l} c(x_i, y_i, f(x_i)) + \lambda_1 \Omega(\|\mathbf{f}\|_{\mathcal{H}}) + \lambda_2 r(x, f)$$

- Loss $c(x, y, f(x)) \equiv |y - f(x)|$
- L1-norm $\Omega(\|\mathbf{f}\|_{\mathcal{H}}) \equiv \|\alpha\|_1 = \sum_i |\alpha_i|$

Design choices

$$\min_{\mathbf{f}\in\mathcal{H}} \sum_{i=1}^{l} c(x_i, y_i, f(x_i)) + \lambda_1 \Omega(\|\mathbf{f}\|_{\mathcal{H}}) + \lambda_2 r(x, f)$$

- Loss $c(x, y, f(x)) \equiv |y - f(x)|$
- L1-norm $\Omega(\|\mathbf{f}\|_{\mathcal{H}}) \equiv \|\alpha\|_1 = \sum_i |\alpha_i|$
- $r(x, f)$ as before

Design choices

$$\min_{\mathbf{f}\in\mathcal{H}} \sum_{i=1}^{l} c(x_i, y_i, f(x_i)) + \lambda_1 \Omega(\|\mathbf{f}\|_{\mathcal{H}}) + \lambda_2 r(x, f)$$

- Loss $c(x, y, f(x)) \equiv |y - f(x)|$
- L1-norm $\Omega(\|\mathbf{f}\|_{\mathcal{H}}) \equiv \|\alpha\|_1 = \sum_i |\alpha_i|$
- $r(x, f)$ as before

$$\min_{\alpha} \quad \frac{1}{l}\sum_{i=1}^{l} |y_i - f(x_i)|_\epsilon + \lambda_1 \|\alpha\|_1 + \lambda_2 \frac{1}{p}\sum_{q=1}^{p} w_q \max(d_q - (f(x_{iq}) - f(x_{jq})), 0)$$

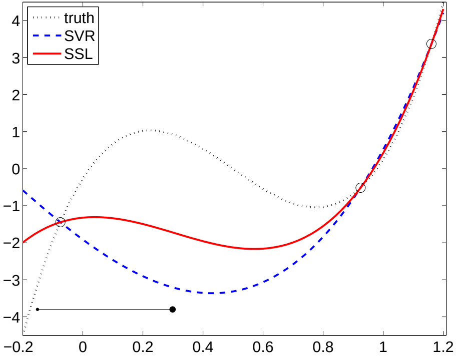
Linear program

Convex, piecewise linear. Convert to a linear program

$$
\begin{aligned}
&\min_{\alpha,\alpha_0,\xi,\eta,\nu} \quad \frac{1}{l}\mathbf{1}^\top\xi + \lambda_1\mathbf{1}^\top\eta + \frac{\lambda_2}{p}\mathbf{w}^\top\nu \\
&\text{s.t.} -\xi - \epsilon\mathbf{1} \le \mathbf{y}_{1:l} - K(\mathbf{x}_{1:l}, \mathbf{x}_{1:l})\alpha - \alpha_0\mathbf{1} \le \xi + \epsilon\mathbf{1} \\
&\quad \xi \ge 0 \\
&\quad -\eta \le \alpha \le \eta \\
&\quad (K(\mathbf{x}^i_{1:p}, \mathbf{x}_{1:l}) - K(\mathbf{x}^j_{1:p}, \mathbf{x}_{1:l}))\alpha \ge \mathbf{d} - \nu \\
&\quad \nu \ge 0
\end{aligned}
$$

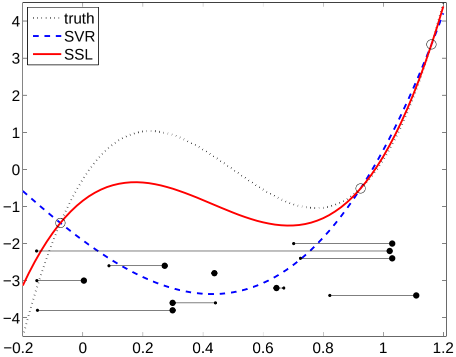
$3l + p + 1$ variables and $5l + 2p$ constraints. Global optimal solution.

A toy example



- True function 3rd order polynomial
- Regression underfits
- Random order preference $f(0.30) - f(-0.15) \geq 0$ improves fit

A toy example



- More random order preferences improve even more

Experiments on benchmark datasets

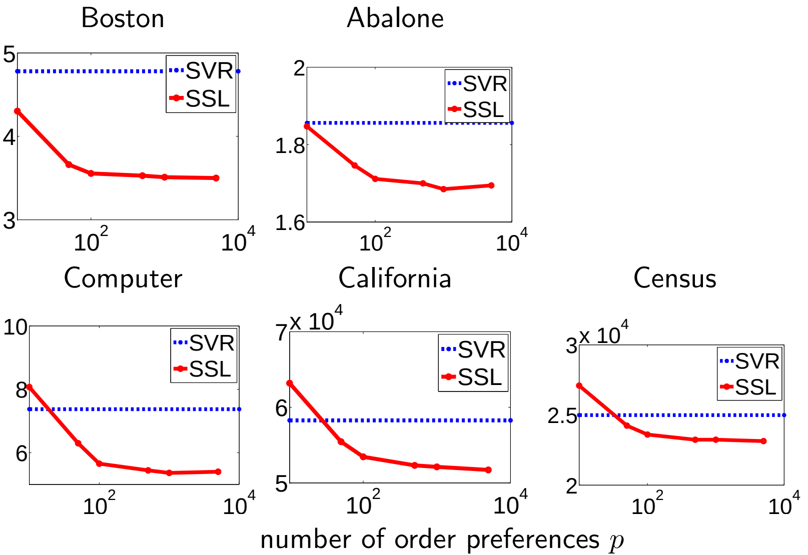
- 5 datasets: Boston, Abalone, Computer, California, Census
- Settings:
  - $w = 1$
  - RBF kernel
  - Kernel bandwidth and λ_1 tuned by CV
  - $\lambda_2 = 1$
  - Test-set error $\sum_{i \in \text{test}} |y_i - f(x_i)| / |\text{test}|$
  - Average over 20 random trials

Oracle order preferences improve regression

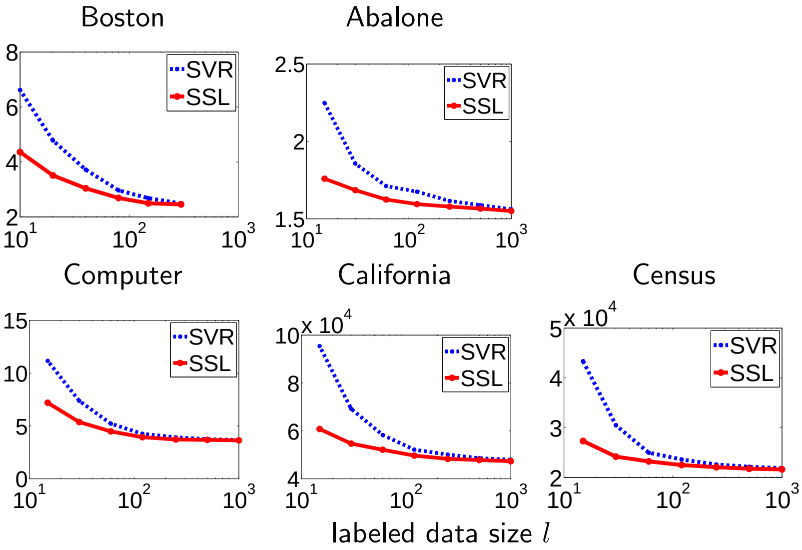
- "Oracle" order preferences $f(x_i) - f(x_j) \geq 0.5(y_i - y_j)$
- 1000 order preferences

Dataset	Partition		Mean absolute error		Improvement
	dim	$l/u/$test	SVR	SSL	
Boston	13	20/200/286	4.780	3.511	27%
Abalone	8	30/1000/3147	1.856	1.685	9%
Computer	21	30/1000/7162	7.373	5.364	27%
California	8	60/1000/19580	58268	52120	11%
Census	16	60/1000/21724	24992	23241	7%

The more preferences, the better

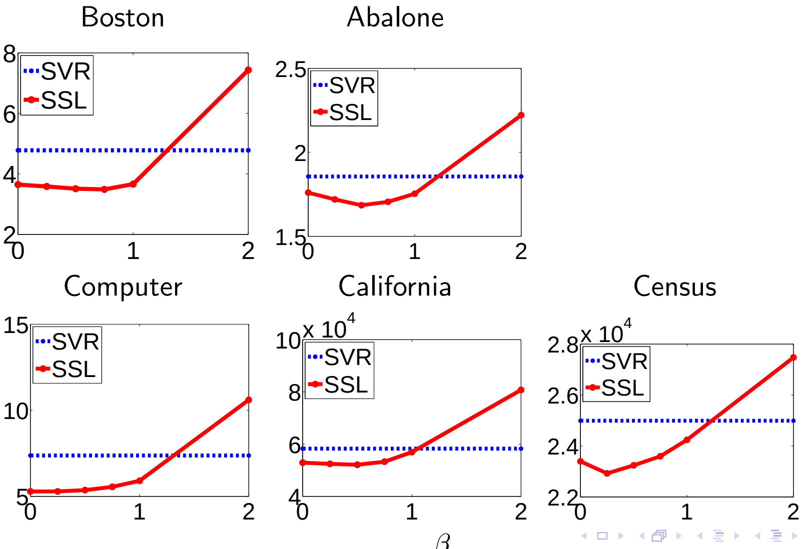


Order preferences most helpful with little labeled data



Order preferences helpful even when imperfect

$$f(x_i) - f(x_j) \geq \beta(y_i - y_j)$$



Experiment with "real" order preferences

Predict house value (California).

Experiment with “real” order preferences

Predict house value (California).

- Roughly equal
  - within 25 miles
  - age difference within 10 years
  - income difference within $1000

Experiment with “real” order preferences

Predict house value (California).

- Roughly equal
  - within 25 miles
  - age difference within 10 years
  - income difference within $1000
- When roughly equal, price ordered by the number of bedrooms.

Experiment with “real” order preferences

Predict house value (California).

- Roughly equal
  - within 25 miles
  - age difference within 10 years
  - income difference within $1000
- When roughly equal, price ordered by the number of bedrooms.
- $d = 0, w = 1, p = 1200$

Experiment with “real” order preferences

Predict house value (California).

- Roughly equal
  - within 25 miles
  - age difference within 10 years
  - income difference within $1000
- When roughly equal, price ordered by the number of bedrooms.
- $d = 0, w = 1, p = 1200$
- 6% reduction in test-set error

Experiment with "real" order preferences

Predict house value (California).

- Roughly equal
  - within 25 miles
  - age difference within 10 years
  - income difference within $1000
- When roughly equal, price ordered by the number of bedrooms.
- $d = 0, w = 1, p = 1200$
- 6% reduction in test-set error
- Post experiment: 30% order preferences were wrong. Robust.

Conclusions

- Order preferences encode domain knowledge
- Linear program for kernel regression
- Even noisy, heuristic order preferences help

Thank you

Questions?

Connection to semi-supervised learning

$$\min_{\mathbf{f}\in\mathcal{H}} \sum_{i=1}^{l} c(x_i, y_i, f(x_i)) + \lambda_1 \Omega(\|\mathbf{f}\|_{\mathcal{H}}) + \lambda_2 r(x, f)$$

Order preferences:

$$r(x, f) = \sum_{q=1}^{p} w_q \max(d_q - (f(x_{iq}) - f(x_{jq})), 0)$$

Graph-based semi-supervised learning (manifold regularization):

$$r(x, f) = \sum_{i,j\in U} w_{ij}(f(x_i) - f(x_j))^2,$$

Semi-supervised support vector machines (S3VMs):

$$r(x, f) = \sum_{i\in U} \max(1 - |f(x_i)|, 0)$$